

修 士 論 文 の 和 文 要 旨

研究科・専攻	大学院 情報システム学研究科 情報システム基盤学専攻 博士前期課程		
氏 名	長塚 郁	学籍番号	1053018
論 文 題 目	複数 GPU 向けの CUDA コードを生成する OpenMP 処理系の研究		
<p>要 旨</p> <p>近年, 複数の GPU(Graphics Processing Unit)を用いた General Purpose computing on GPUs (GPGPU)が一般的となった. CPU と比較して非常に高い並列度と高いメモリバンド幅を持つ GPU を複数個用いることによって, 気象シミュレーションや計算の高速計算が行われるようになった. しかし, Nvidia 社の統合開発環境 CUDA(Compute Unified Device Architecture)により GPGPU プログラムの敷居は下がったものの, 複数 GPU 向けの GPGPU プログラムは特有のアーキテクチャの知識の習得や手間のかかるプログラミングを行って初めて高い性能が得られるため, 一般のプログラマに敬遠されてしまう傾向にある.</p> <p>そこで本研究では OpenMP プログラムから複数 GPU 向けの CUDA プログラムを生成する OpenMP 処理系を開発することで, 複数 GPU を用いた GPGPU プログラミングの簡易化を図る事を目的とする. この生成を行うのに必要な機能として, 以下の二つが挙げられる.</p> <ul style="list-style-type: none"> ・ OpenMP のメモリモデルと複数 GPU 向け CUDA のメモリモデルとの対応付け ・ OpenMP の並列処理モデルと複数 GPU 向け CUDA の並列処理モデルとの対応付け <p>メモリモデルの対応付けは CPU のメインメモリを各 GPU の共有メモリと見立てることで対処することとした. 並列処理モデルの対応付けは各 GPU へのイタレーション割り当てをブロック分割で行い, GPU Thread と GPU Block への割り当てを実行時ライブラリによって行う事で対処することとした.</p> <p>本研究で提案する OpenMP 処理系 “OMPCUDA-for-MultiGPU” は Omni OpenMP Compiler を基に開発された OMPCUDA を複数 GPU 向けに拡張することで実装した.</p> <p>OMPCUDA-for-MultiGPU で生成を行ったプログラムの評価の結果, 行列積計算プログラムではハンドコーティングによる CUDA プログラムにほとんど近い実行時間を得られ, 有効性を検証できた. 一方, グレゴリ級数による円周率演算プログラムでは, 行列積計算プログラムほどハンドコーティングによる CUDA プログラムに近い実行時間を得られず, その原因が GPU デバイスメモリの宣言であるのが分かった. その処理による実行時間への影響が大きい問題について, 格子点の近傍計算の並列ループの実験を行うと, 1 個の GPU を用いた場合の実行時間が, 2 個, 4 個の GPU を用いた場合の実行時間よりも短くなる. そこで, OMPCUDA-for-MultiGPU が生成する CUDA プログラムの改善案を提案し, ハンドコーティングして実行したところ, 複数の GPU を用いたほうが単一 GPU よりも実行時間が短くなった.</p>			

